

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s):	F. AHMAD et al.	Examiner	Kamal B. Divecha
Serial No.	09/972,362	Group Art Unit	2151
Filed	October 5, 2001	Docket No.	SJO920010099US1
TITLE	METHODS AND APPARATUS FOR LAUNCHING DEVICE SPECIFIC APPLICATIONS ON STORAGE AREA NETWORK COMPONENTS		

PRE-APPEAL BRIEF REQUEST FOR REVIEW ARGUMENTS

Applicants request review of the Examiner rejection of claims 1, 2, 4-9, and 21-40 as obvious (35 U.S.C. §103) over Weber (U.S. Patent No. 6,480,901) in view of Ismael (U.S. Patent No. 6,851,118) in the Final Office Action dated September 4, 2007 (“FOA”).

With respect to claims 1, 21, 24, and 31, Applicant’s request review of the Examiner’s finding that col. 16, line 51 to col. 17, line 51 of Weber teaches the claim requirement of receiving selection of one of the displayed application processes residing on the selected network component (FOA, pg. 5)

The cited col. 16, line 51 to col. 17, line 51 of Weber discusses how upon discovering a list of devices, to start a management interface application, the user preferably double clicks on a storage system and the device property information about the selected storage system is received. The device properties include the storage system’s management interface version and the management interface application program version. The management interface application program is loaded on the management station and then may be used to change the configuration of one of the devices.

The cited cols. 16-17 do not teach receiving selection of a displayed application process residing on a selected network component. Instead, the cited cols. 16-17 discuss how a user selects (double clicks) on a device and receives properties that include a management interface application that is loaded on the management workstation to change the configuration of the device. The cited management interface application is not an application process on the selected network component as claimed that the user may invoke using the communication interface type and parameter name in the rules file. Instead, the cited management interface application is a program that is locally run to manage the remote device.

Applicants request review of the Examiner’s finding that col. 13, lines 1-49 col. 7, lines 25-39; col. 16, lines 58-67 of Weber teaches the claim requirement of launching the selected

application process on the selected network component using the determined communication interface from the rules file. (FOA, pg. 6)

The cited col. 13 discusses a discover-monitor application screen having a management domain window presenting a tree view of the management domain. Lower level nodes in the tree represent actually physical hardware devices such as servers, arrays, and other I/O devices. The higher level nodes in the tree represent the location of the hardware devices, such as state and city. A detailed information window presents detailed properties for each device. If a device is selected, the device's management interface application program is launched. The cited col. 13 discusses a display of hardware devices in the network and their properties. The cited col. 16 discusses starting a management interface application for a storage system in the network by the user clicking one of the storage systems. The device property information about the selected storage system is received. The cited col. 7 discusses how the management application for the device communicates with the controller and control software of the device.

Nowhere do the cited cols. 7, 13, and 16 of Weber anywhere teach or suggest the claim requirement of launching the selected application process on the selected network component using the determined communication interface type and parameter name from the rules file. Instead, the cited Weber discusses how to launch a management application interface on the local management workstation to manage, monitor and configure the device on the network. There is no teaching of using a communication interface type and parameter name to launch a selected application process on the selected network component as claimed.

Applicants submit that Weber discusses how to determine a management application to run locally to control software on a remote device. This does not teach or suggest the claim requirement of using a communication interface type and a parameter name to invoke an application residing on the network component. Weber does not use information on the device to invoke an application residing on the network, but instead uses the information to invoke a management application to communicate with the remote device and application.

Further, the claims require two levels of selection, one of the displayed network component and another of one application process associated with the network component. The cited Weber discusses that when a device node is selected, the device's management interface application is launched (Weber, col. 13, lines 44-50). This does not teach or suggest, first selecting a device and then displaying application processes residing on the network component,

and then receiving selection of one displayed application process on the network component. Instead, the cited Weber discusses receiving selection of a device and then going straight to launching on the management workstation the management application interface for that device.

The Examiner found that Weber does not teach the claimed rules file. Applicants request review of the Examiner finding that col. 7, line 44 to col. 8, line 67 of Ismael teaches the claimed rules file (FOA, pg. 7), which requires a rules file having at least one rule for each of the network components, wherein each rule identifies the network component to be managed, one of a plurality of communication interface types, and a parameter name, wherein the parameter name is used with the communication interface type to invoke the application process residing on the network component.

The cited cols. 7-8 of Ismael discuss a managed object adaptor server that abstracts a communication protocol and enables applications to perform management operations on a network system agent, such as to allow the agent to be queried by remote applications that use different protocols. Different managed object adaptor servers may be provided for different protocols, such that the agents may be managed using different communication or management protocols.

Nowhere do the cited cols. 7-8 of Ismael teach or suggest a rules file having, for each of a plurality of network components, a communication interface and parameter that are used to launch the application process on the network component. Instead, the cited cols. 7-8 of Ismael discuss the use of the managed adaptor server for one system agent and to perform management operations for the agent. There is no teaching that the managed object adaptor server have communication and parameter information for multiple network components to launch the remote network components.

The Examiner further cited col. 9, lines 30-59, col. 18, lines 20-60, col. 1, line 50 to col. 2, line 41 of Ismael. (FOA, pg. 7). The cited col. 9 mentions an adaptor API to enable Java management applications to communicate with network management system agents. A managed object adaptor enables Java management applications access to management objects. The cited col. 18 mentions that to initialize an adaptor, a client application invokes a connection method. The cited parameters relate to the host name of the agent, port number and logical name which is dependent on underlying communication mechanism. Again, although the cited Ismael discusses an adaptor that provides a communication protocol to access a management object, this does not

teach a rules file providing communication protocols for different network components used to allow launching of application processes on network components.

Applicants submit that even if one were to combine Weber and Ismael, the proposed combination does not teach or suggest the combination of requirements of first selecting a device, displaying application processes residing on the network component, and then receiving selection of one displayed application process on the network component to launch, where a rules file identifies for each network component, a communication type and parameter name used to invoke the application process residing on the network component. Instead, the proposed combination provides the launching of a local management application interface to interface with a device and a managed adaptor server for one system agent that is used to perform management operations for a remote agent. In both cases, Weber and Ismael provide a local tool used to interface with a remote agent or device. This does not disclose the claim requirements of using a rule to access a network component and then select an application process residing on the selected network component to launch the selected application process on the network component.

With respect to claims 2, 22, 25, 32, Applicants request review of the Examiner's finding that col. 13, lines 1-67, col. 4, lines 34-52, col. 9, lines 10-42, col. 16, line 51 to col. 17, line 35 of Weber as teaching the claim requirement of a graphical output device coupled to the interface process for displaying one or more graphical objects representing the application processes on the network components. (FOA, pgs. 7-8).

The cited sections of Weber discuss the display of network components. For instance, cited col. 13 and FIG. 6 show components, such as storage arrays and device nodes. The cited col. 4 discusses display of I/O devices, the cited col. 9 discusses an applet for discovering devices and a user interface for invoking the management interface to manage the devices. Nowhere do these cited sections teach or suggest graphical objects representing application processes on network components. Instead, the cited Weber discusses displaying information on devices and a management interface that may be used at a local workstation to manage a remote device.

Dated: December 4, 2007

By: /David Victor/
David W. Victor
Registration No. 39,867